НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ ЗАХИСТУ ІНФОРМАЦІЇ

«До захисту допущено»
В.о. завідувача кафедрою
_____ М.М.Савчук
(підпис) (ініціали, прізвище)
“___”_____20 _ р.

Дипломна робота

на здобуття ступеня бакалавра

з напряму підготовки : 113 «Прикладна математика»
(код і назва)

на тему: Обґрунтування стійкості геш-функцій, зручних для використання у SNARK-доведення _____
_____

Виконав: студент 4 курсу, групи ФІ-62
(шифр групи)

Кучерявий Костянтин Ігорович _____ _____
(прізвище, ім’я, по батькові) (підпис)

Керівник Професcор кафедри ММЗІ, Ковальчук Л.В _____ _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант _____ _____ _____
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент_____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ – 2020 року

Національний технічний університет України
«Київський політехнічний інститут
імені Ігоря Сікорського»
Фізико-технічний інститут

Кафедра математичних методів захисту інформації

Рівень вищої освіти – перший (бакалаврський)

Напрям підготовки - 113 «Прикладна математика»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедрою

М.М.Савчук

_____
(підпис) (ініціали, прізвище)

«___»_____20__ р.

ЗАВДАННЯ
на дипломну роботу студенту

Кучерявий Костянтин Ігорович
(прізвище, ім'я, по батькові)

1. Тема роботи Обґрунтування стійкості геш-функцій, зручних для використання у SNARK-доведення,

керівник роботи Ковальчук Людмила Василівна д.т.н. проф,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від _____ р. № _____

2. Термін подання студентом роботи _____

3. Вихідні дані до роботи _____
_____
_____
_____

4. Зміст роботи Обґрунтування стійкісті геш-функції Poseidon до диференціального та лінійного криптоаналізу, при різному виборі S-блоків у базовому алгоритмі блокового шифрування HADES.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо) _____

_____
_____
_____

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1.	Визначення напрямку дослідження	1.09-1.15	
2.	Опрацювання робіт по темі	1.16-1.02	
3.	Узгодження теми	1.02-3.03	
4.	Аналіз протоколів доведення з нульовим розголошенням	3.03-16.03	
5.	Аналіз блокового алгоритму HADES	16.03-2.04	
6.	Аналіз геш-функції Poseidon	3.04-1.05	
7.	Застосування лінійного та диференціального криптоаналізу	02.05-25.05	
8.	Формулювання результатів дослідження	26.05-28.05	
9.	Формування результатів	28.05-04.06	

Студент _____ Кучерявий К.І
(підпис) (ініціали, прізвище)

Керівник роботи _____ Ковальчук Л.В.
(підпис) (ініціали, прізвище)

РЕФЕРАТ

Кваліфікаційна робота містить: 38 стор., 4 рисунки, 18 джерел. У нашій роботі ми описали проблему анонімності у криптовалютах та розглянули вже запропоновані варіанти покращення рівня анонімності. Зокрема, протоколи з нульовим розголошенням zk-SNARK, які досить широко використовуються у питаннях пов'язаних з анонімністю. Також ми детально розглянули нову геш-функцію Poseidon, яка була сконструйована таким чином, щоб якнайкраще підходити для роботи саме з протоколами без розголошень та з використанням блокового шифру HADES. Ми оцінили криптографічну стійкість блокового шифрування HADES, на базі якого була побудована геш-функції Poseidon, до диференціального та лінійного криптоаналізу, при різному виборі S-блоків.

ZK-SNARK, HADES, ГЕШ-ФУНКЦІЯ POSEIDON, ЛІНІЙНІ АТАКИ, РІЗЦЕВІ АТАКИ, АНОНІМНІСТЬ

ABSTRACT

In our work, we have described the problem of anonymity in cryptocurrency, and considered options to improve the level of anonymity. In particular, zk-SNARK (zero knowledge proof), which is widely used in issues related to anonymity. We also considered in detail a new hash function Poseidon, which was designed to be as good as possible to work with protocols with zero knowledge proof, and using the block cipher HADES. We evaluated the cryptographic robustness of the Poseidon hash function, in differential and linear cryptanalysis, with different selection of S-blocks in the basic HADES block cipher algorithm.

ЗМІСТ

ВСТУП

Актуальність дослідження. Актуальність даного дослідження полягає у тому, що ми працюємо та досліджуємо відносну нову геш-функцію Poseidon, запропоновану у 2019 році. Вона пропонується для роботи з протоколом доказу без розголошення - zk-SNARK, який використовують більшість криптовалют, що намагаються забезпечувати реальну анонімність. Обґрунтування стійкості цієї геш-функцій є досить важливим. Адже автори Poseidon, у своїй роботі опирались на аналіз стійкості алгоритму блокового шифрування HADES, до різних типів атак, зокрема алгебраїчних та статистичних. Але, результати стосовно стійкості Hades до криптоаналізу виявились помилковими. актуальність нашого дослідження і отримання реальних оцінок стійкості, є досить великою.

Метою дослідження є оцінка та формулювання висновку, щодо перспективності використання геш-функції Poseidon, Для цього нам необхідно розв'язати **задачу дослідження**, яка полягає в оцінці стійкості цієї геш-функції, до різних типів атак Для розв'язання задачі необхідно вирішити такі завдання:

1) провести огляд опублікованих джерел за тематикою дослідження;

2) сформулювати теоретичне підґрунтя, необхідне для глибокого розуміння основних принципів та ідей закладених у цій геш-функції;

3) дослідити методи криптоаналізу, та проаналізувати функцію Poseidon;

4) провести математичні обрахунки, та отримати математично обгрунтувати стійкість геш-функції Poseidon.

Об'єктом дослідження надійність досліджуваної геш-функції застосовної у zk-SNARK доведеннях.

Предметом дослідження є моделі та методи диференціального та лінійного криптоаналізу симетричного блокового шифру HADES, за допомогою якого створена геш-функція.

При розв'язанні поставлених завдань використовувались такі *методи дослідження*: методи диференціального криптоаналізу, методи лінійного криптоаналізу, елементи комбінаторики.

Наукова новизна отриманих результатів полягає у конкретній, та математично обґрунтованій оцінці надійності запропонованих S-блоків (Блоків підстановки), що використовуються у геш-функції Poseidon, та оцінка надійності S-блоків які ще не були попередньо описані у роботах .

Практичне значення результатів полягає у тому, що ми отримаємо конкретне практичне значення кількості раундів, необхідних для застосування, для гарантування стійкості геш-функції Poseidon.

1 АНОНІМНІСТЬ У КРИПТОВАЛЮТАХ

Починаючи зі створення першої широко відомої криптовалюти Bitcoin[1] за протоколом Накамото, поняття анонімності та криптовалют завжди йдуть поруч. Адже багато криптовалют на початку свого створення заявляли про надання анонімності для своїх користувачів. Анонімність у фінансовому полі, дійсно є досить суперечливим поняттям. Адже у більшості звичайних людей та бізнесів існує бажання, а у країнах з незрілими демократичними інституціями часто навіть необхідність, зберігати анонімність своїх фінансів. І це не завжди тому, що вони замислили щось протизаконне, а здебільшого через те, що вони самі не бажають стати жертвами протиправних дій зловмисників. Напротивагу цього, уряди практично всіх технологічно розвинених країн намагаються встановити тотальний моніторинг та контроль над обігом коштів як своїх громадян. У першу чергу це викликано задачами протидії відмиванню коштів, несплати податків та загрозам фінансування тероризму, але існують також інші численні причини інтересу держави до цього питання. А 10 січня 2020 року вступила в сильну П'яту директиву Проти відмивання Грошей (5AMLD) [2], , яка була прийнята в ЄС у травні 2018 року. Директива, говорить про те, що в уряди мають забезпечити в обов'язковому порядку, перевірку всі клієнтів криптовалютних платформм. Незважаючи на контраверсійність цього питання, важко заперечувати що останнім часом, у більшості людей, зростає бажання зберегти інкогніто.

Способи збереження анонімності джерела надходження коштів Кожна криптовалюта, за замовчуванням має певний рівень анонімності. Який у загальному випадку, звісно є значно кращим ніж у централізованих банківських систем. Адже майже всі криптовалюти для реєстрації гаманця не потребують особистих даних користувача, але тільки у тому випадку, якщо користувач створює гаманець прямо, не

використовуючи криптовалютні біржі. Більш коли користувач створює довільну кількість гаманців так, що неможливо буде визначити, чи належать вони одній особі, чи різним. Більш того, сучасні програми-клієнти можуть автоматично створювати для користувача нові адреси гаманців для кожної транзакції, щоб його було складніше відстежити

Але, напротивагу цьому, за своєю природою блокчейн передбачає що інформація про всі транзакції назавжди зберігаються у ньому,і ця інформація є загальнодоступною, і такою, що не видаляється. А якщо ви, створюєте гаманець на біржі, то зберігати анонімність ще важче. Адже біржі зобов'язані відповідати вимогам законодавства тієї країни, до якої належить ця біржа, і вони вимагають певну особисту інформацію про клієнта.

1.1 Можливі шляхи покращеня анонімізації у криптовалютах

Наведемо деякі варіанти того, як можна покращити реально анонімізацію.

Самостійний майнінг криптовалют Найпростіший спосіб отримувати криптовалюти з анонімного джерела – це майнити їх самому. З недоліків цього способі можна виділити наступні, по-перше, накопичення монет у такий спосіб відбувається досить довго. По-друге, він вимагає постійної затрати ресурсів (електроенергії) та постійного підключення до інтернету.

Використання міксерів транзакцій Загальний принцип робота міксера транзакцій полягає у наступному. На одному з етапів пересилання коштів виконується так звана колективна транзакція (або групова транзакція), в якій беруть участь відразу кілька платників. Монети, які вони пересилають, розбиваються на дрібні "шматочки які

перемішуються між собою, і відправляються отримувачам за певним алгоритмом. В результаті отримувачі отримують правильну кількість монет, але ця сума буде складатись з різних шматочків, відправлених різними платниками. Механізм анонімізації повинен приховати однозначну відповідність між відправниками та їх монетами. В результаті для отримувача та для сторонніх спостерігачів відправник коштів стає невизначеним. Невизначеність полягає у тому, що отримані кошти могли, з рівною імовірністю, бути відправленими будь-яким з учасників колективної транзакції. Відповідно, чим більша кількість платників бере участь у колективній транзакції, тим з меншою імовірністю можна вгадати фактичного відправника.

Саме за таким принципом влаштовані BTC-міксери. Серед недоліків цього способу, можемо виділити наступні, по-перше, закони деяких країн, забороняють використання таких міксерів. Та також, іноді може виникати досить суттєва затримка у часі доставки монет.

Використання анонімних платіжних систем Альтернативами міксерів можуть бути спеціальні гаманці з великим ступенем анонімності, наприклад, Electrum, чи платіжна система Z-Pay з відкритим вихідним кодом. При перетворенні криптовалютних коштів на фіатні і виведенні їх на карту через Z-Pay це виглядає як поповнення через термінал або як переказ від фізичної особи. Але для реєстрації у системі потрібно ввести електронну адресу та номер телефону, або виконати вхід через соціальну мережу. Тож анонімність не можна назвати повною.

Купівля-продаж криптовалют через криптомати або біржи без верифікації. Найпростіший спосіб купівлі біткоїнів – через криптомат (спеціальний апарат для купівлі криптовалют, схожий на банкомат). Більшість криптоматів, крім BTC, підтримують також Ethereum, Bitcoin Cash, Litecoin, Dash, Dogecoin, а також анонімні Zcash и Monero, а деякі навіть більше 40 різних криптовалют. Процедура купівлі дуже проста, але й комісія досить висока: 5-6% від суми операції з гривни та + 0.001 BTC комісія самої біткоїн-мережі. В Україні криптомати вже

існують у кількості 8 штук. Деякі “високорівневі” моделі криптоматів, наприклад, від компанії Genesis Coin, дозволяють не лише купувати криптовалюту, а й обмінювати її на фіатні гроші. На відміну від купівлі криптовалюти, її продаж через криптовалютний АТМ займає більше часу, оскільки відбувається у два етапи. Користувачу потрібно виконати наступні дії. 1. Вказати суму криптовалюти, яку він планує продати. 2. Пройти процедуру верифікації (за потребою), на якій ми зупинимось нижче. 3. Відсканувати QR-код на екрані пристрою або на виданому чеку для переводу вказаної суми на гаманець оператора (час, відведений на цю процедуру, обмежений). 4. Зачекати підтвердження транзакції нодами мережі криптовалют (як правило, достатньо двох підтверджень). 5. Відсканувати QR-код на видачу готівки, після чого криптомат видасть фіатні гроші. При обміну криптовалюти на фіатні гроші криптомат також отримує комісію. Слід зазначити, що комісія, яка втримується криптоматами, має тенденцію до зниження через зростаючу конкуренцію між операторами. Ще кілька років тому комісія складала 10-12%, а сьогодні — 5-6%. Зупинимось детальніше на проблемі анонімності та можливої верифікації. По-перше, виходячи з опису протоколу роботи криптомату, він пов’язаний з деяким криптовалютним гаманцем Management у "internal wallet"на 2 хв 19 сек). За адресою цього гаманця, взагалі кажучи, можливо відслідкувати цей криптомат як джерело поповнення коштів. І хоча особа того, хто вносив кошти, лишається невідомою, проте можна виявити його географічне положення. Щодо анонімності отримувача коштів, то тут можливі різні варіанти. При обміні криптовалюти на фіатні гроші через криптомат верифікація може бути потрібною у у двох випадках, або сума до обміну перевищує встановлений ліміт (проте в більшості таких випадків оператори просто не дозволяють обмінювати суми, які перевищують ліміт, шляхом встановлення обмежень; або вимоги обов’язкової верифікації, що встановлені в певних країнах, незалежно від суми операції. Процедура верифікації може виконуватись різними способами, наприклад, шляхом

фотографування з певним підтверджуючим документом у руках, або скануванням відбитків пальців, або шляхом зчитування даних платіжної карти, або через номер телефону.

1.2 Принципи функціонування анонімних криптовалют, їх переваги та недоліки

Анонімні монети — це пірінгові (з'єднання кожен з кожним) платіжні системи з власною внутрішньою розрахунковою одиницею. Їх головна мета — забезпечити повну конфіденційність фінансових операцій за допомогою спеціальних технологій та криптографічних протоколів.

Безумовно анонімна криптовалюта Monero. Розробник анонімної криптовалюти Monero Рікардо Спаньї вважає, що цифровій індустрії необхідна фінансова приватність. В іншому випадку, ринок буде заповнений таргетованою (направленою) рекламою, а в гіршому – почнуться злочини, направлені на власників значної суми криптовалют (адже з "традиційного"блокчейну кожен може дізнатися баланс будь-якого чужого біткоїн-гаманця). Monero — це поки єдина монета, всі операції з якою є анонімними за замовчуванням [3]. Для всіх інших криптовалют функцію анонімності потрібно додатково налаштовувати (як BTC-міксери) або підключати при проведенні транзакції (як у криптовалюті Dash). З публічного блокчейну Monero можна дізнатись, чи існує та чи інша адреса рахунку, але неможливо перевірити баланс цього рахунку та отримати доступ до історії транзакцій, з ним пов'язаних. Анонімність Monero забезпечується наступними складними криптографічними засобами та механізмами: - кільцевими конфіденційними транзакціями з використанням доведення діапазону та криптографічними коммітментами (підтвердженнями), які дозволяють приховувати величини коштів, що пересилаються у транзакції, - неінтерактивними доведеннями без розголошень, які дозволяють

перевіряти транзакцію без знання відправника та отримувача коштів, без знання кількості монет та без використання довіреної сторони; - одноразовими адресами, та одноразовими відкритими ключами, які користувач повинен генерувати за допомогою криптографічних механізмів та які приховують шлях пересування коштів; - кільцевими підписами, що дозволяють приховати, з якої саме адреси (з певної групи, що складається з 11 адрес) було відправлено кошти. Ще однією безумовною перевагою є відкритий код цієї криптовалюти та високий ступінь довіри до її розробників у криптовалютному суспільстві. До недоліків Monero можна віднести високі комісії за транзакції.

Криптовалюта ZCash Zcash [4] була розроблена компанією Zerocoin Electric Coin Company та анонсована 20 січня 2016 року. Це криптовалюта з відкритим вихідним кодом, що безумовно є її перевагою. Вона забезпечує конфіденційність та часткову прозорість транзакцій. Мається на увазі, що всі транзакції Zcash публікуються у загальнодоступному ланцюжку блоків, але відправник, отримувач та сума транзакції залишаються приватними. для забезпечення анонімності транзакції вперше застосувала доведення без розголошення нового типу, яке називається *zk-SNARK* [5]. Абревіатура*zk-SNARK* розшифровується як “zero knowledge Succinct Non-interactive ARguments for Knowledge”, тобто "не інтерактивні стиснені аргументи знання з нульовим розголошенням". Термін "не інтерактивні"означає, що сторона, яка доводить знання певної інформації, будує це доведення самостійно, без взаємодії з перевіряючою стороною. Вислів "з нульовим знанням"означає, що сторона, яка доводить знання деякої секретної інформації, будує такі аргументи для перевіряючого, які мають наступні властивості: - з одного боку, повністю (з використанням складного математичного апарату) переконують перевіряючого у тому, що ця інформація їй відома; - імовірність переконати перевіряючого без знання цієї інформації практично дорівнює нулю; - з іншого боку, доведення знання цієї інформації не дає ніякої інформації про сам секрет. Технологія zk-SNARK

є досить багатообіцяючою, і ми її розглянемо детальніше у наступному розділі. Цей протокол, звісно не є єдиним протоколом з нульовими розголошеннями, також варто виділити такі протоколи як zk-STARK [6]., та Bulletproofs [7]

Висновки до розділу 1

Питання анонімності дійсно є доволі актуальним в даний момент часу для усього суспільства, яке використовує криптовалюти. На сьогодні Monero, Dash та ZCash є найбільшими за ринковою капіталізацією приватними цифровими валютами. Всі вони показують стрімкий ріст зацікавленості до них, що говорить про реальну цінність ідеї анонімності. Спільною рисою всіх цих криптовалют є той факт, що всі вони використовують у своїй роботі протоколи доведення з нульовим розголошенням. Зокрема, протокол *zk-SNARK*, який є одним з найпопулярніших у даний момент. Проте, це далеко не єдиний протокол з нульовим розголошенням. В свою чергу, популярність такого виду протоколів викликає необхідність створення нових геш-функцій, за допомогою яких вони будуть працювати. Одну з таких геш-функцій ми розглянемо більш детально у наступному розділі.

2 ПОБУДОВА ГЕШ-ФУНКЦІЇ POSEIDON, ТА ДОПОМІЖНИХ КОНСТРУКЦІЙ.

У даному розділі, ми розглянемо принцип побудови геш-функції POSEIDON. Задля цього, ми розглянемо дві конструкції, на базі яких, ця функція побудована. А саме, блоковий алгоритм HADES та застосування функція губки (англ. sponge construction) для побудови геш-функцій.

Також, ми розглянемо більше детально протокол SNARK, який використовується у застосуванні геш-функції POSEIDON

2.1 Протоколи доведення з нульовим розголошенням

Розглянемо найбільш популярний протокол цього виду, а саме, **zk-SNARK** (англ. Zero-Knowledge Scalable Transparent Argument of Knowledge) В перекладі, короткий неінтерактивний аргумент знання з нульовим розголошенням. Основна його ідея, була вже наведена у 1 розділі. На цей час, є найбільш широко використовною системою доказу, з нульовим розголошенням. Доказ з нульовим розголошенням дозволяють одній людині доводити іншій, що твердження є вірним, не розкриваючи будь-якої інформації, що виходить за рамки дійсності твердження. Залучені сторони зазвичай називають як *перевіряючий* і *верифікуйючий*, а твердження, яке вони тримають в секреті, називають *свідком*. Основна мета цих доказів полягає в тому, щоб розкрити якомога менше даних між двома сторонами. Іншими словами, можна використовувати докази з нульовим розголошенням, для підтвердження того, що вони мають певні знання, не розкриваючи жодної іншої інформації.

Акронім SNARK "стислий"означає, що ці докази менше за розміром і можуть бути швидко перевірені. "Неінтерактивний"означає, що взаємодія між *перевіряючим* і *верифікуйючим*, практично відсутня.

Більш старі версії протоколів з нульовим розголошенням зазвичай вимагають, щоб перевіряючий і верифікуйте контактували між собою і тому вони вважаються «інтерактивними» доказами без розголошення. А в «неінтерактивних» конструкціях перевіряючий і верифікуйте повинні обмінюватися тільки одним доказом.

Її робота заснована на використанні схеми *R1CS*. Обчислення відбуваються у вигляді арифметичних дій, множення та додавання над скінченним полем $GF(p)$. Складність генерації доказів прямо пропорційна кількості обмежень.

2.2 Блоковий алгоритм HADES

У роботі [8] був представлений новий блоковий алгоритм HADES.

Опишемо основну ідею цієї роботи. Автори використовують стратегію "широкого шляху"Блокові шифри та криптографічні перестановки, як правило, розроблені шляхом багаторазового повторення ефективно реалізованої функції, з надією, що отриманий результат поводитися як випадково рівномірно розподілена перестановка. Загалом, ця ж раундова функція достатньо разів повторена, щоб переконатися, що будь-які симетрії та структурні властивості, які можуть існувати в раундової функції, зникають. У нашому випадку, замість того, щоб розглядати одну і ту ж раундову функцію для побудови шифру ми пропонуємо розглянути змінну кількість S-блоків на раунд, тобто , використовувати різні шари *S-Box* у раундових функціях. Кожен раунд шифру на основі HADES складається з трьох етапів:

1) Додавання раундового ключа (англ. Add Round Key) позначаємо як ARK(·);

2) Підстановка (англ. SubWords) позначаємо як SB(·);

3) Замішування шару (англ. MixLayer) позначаємо як M(·).

Схематичний опис цього алгоритму можемо побачити на рисунку 2.1

Як можемо бачити, на останньому етапу відбувається останнє

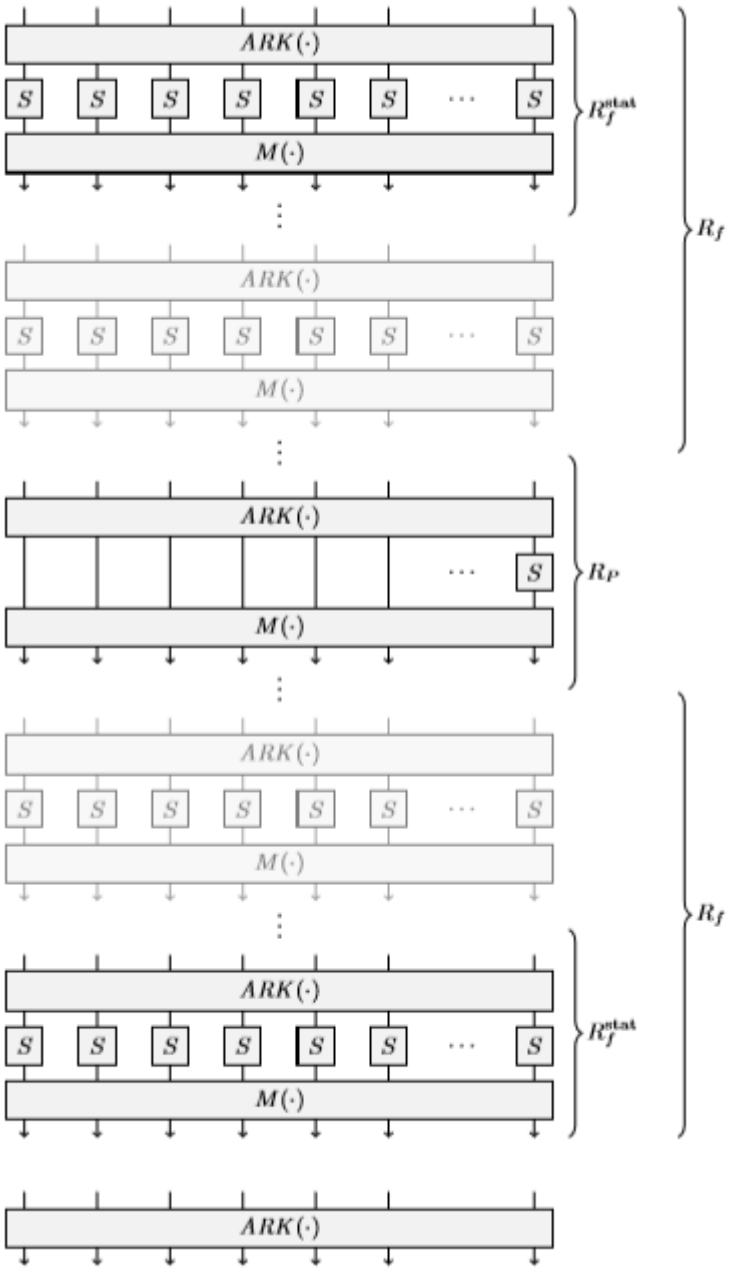

Рисунок 2.1 – Конструкція Hades

раундове додавання, а операція змішування шару пропущена.

$$\underbrace{ARK \to SB \to M}_{\text{1 раунд}} \to ... \to \underbrace{ARK \to SB \to M}_{\text{R-1 раунд}} \to \underbrace{ARK \to SB}_{\text{R раунд}} \to ARK$$

Ключовою властивістю Hades є те, що кількість S-блоків на раунд не однакове для всіх раундів:

– Деяке визначене число раундів, позначене R_f має повний шар S блоків, позначимо їх кількість як t.

– Деяке визначене число раундів, позначене R_p має не повний шар S блоків, тобто лише $1 \leq s < t$ перетворень. Та $(t - s)$ тотожних відображеннь.

У нашому випадку, автори розглядають схему тільки з $s = 1$. Маємо що, R_p раунди мають лише один блок заміни за раунд, та $(t-1)$ тотожних перетворень.

Розглянемо цю схему більш детально, маємо $R_F = 2*R_f$ яке є парним числом. Тоді:

– Перші R_f раундів мають повний шар S-блоків.

– Проміжні R_p раундів мають частковий шар S-блоків. (у нашому випадку, один S-блок)

– Останні R_f раундів мають повний шар S-блоків.

Стратегія HADES не дає жодних обмежень щодо вибору лінійного шару та щодо вибору S-блоків. Як вже було сказано, головна ідея полягає в тому, щоб розглянути "традиційний"шифр SP, заснований на стратегії широкого шляху, а потім замінити певну кількість раундів з повним шаром S-блоків на однакову кількість раундів з не повним шаром S-блоків, щоб мінімізувати кількість нелінійні операції, але без впливу на криптографічну стійкість.

2.3 Функція губки

В криптографії функція губки (англ. sponge construction) була вперше введена у роботі [9]. Вона належить до класу алгоритмів із кінцевим внутрішнім станом, на вхід якого надходить двійковий рядок довільної довжини, і який повертає двійковий рядок також довільної довжини.

Функція губки - це ітеративна конструкція для створення функції F з довільною довжиною на вході й довільної довжини на виході на основі перетворень f. Губка має внутрішній стан S — з даними фіксованого розміру b (кількість біт). Значення b називається ширина. При цьому дані розділені на 2 частини — перша S_1 розміру r, а друга S_2 — розміру c. Значення r називається бітовою швидкістю, а значення — потужністю. $b = r + c$

Спочатку, блок даних розміру b заповнюється нулями, а вхідні дані M розбиваються на блоки розміру r, На рис 2.2 схематично зображено роботу функції губки. Подальша робота губки проводиться в 2 етапи:

1) У фазі *вбирання* (англ. absorbing), над $r-$бітами входу, виконується операція XOR ($\oplus$), перетворюючи їх у перші $r-$біти стану. Після чого, застосовується функція f доки, доки не будуть вичерпані всі блоки вихідного повідомлення. Далі відбувається настає наступна фаза.

2) У фазі *вичавлювання*(англ. squeezing),функція f застосовується до перших r-бітів стану та повертаються у вигляді вихідних блоків. Кількість вихідних блоків вибирається за бажанням користувача.

Останні c бітів залежать від вхідних блоків лише опосередковано, й не виводяться в ході фази «вичавлювання».

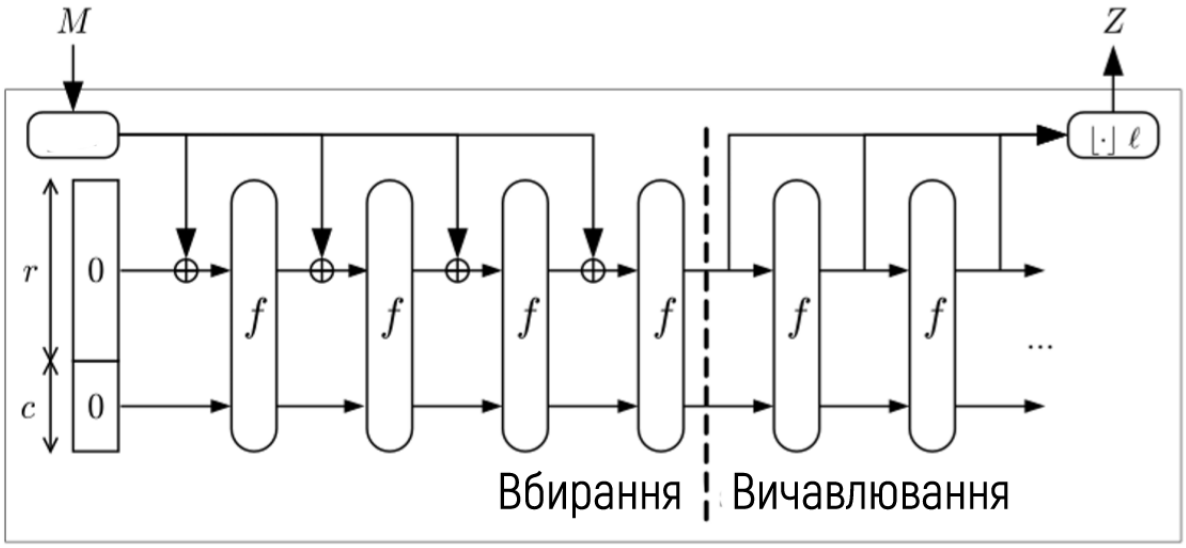

Рисунок 2.2 – Функція губки з двома етапами

2.4 Геш-функція Poseidon

Зрештою, у 2019 році, у роботі [10] була представлена нова геш-функція, під назвою POSEIDON. Poseidon - це хеш-функція, розроблена для мінімізації складності перевірки та верифікації при створенні та підтвердженні доказів з нульовим знанням. Порівняно з конкурентами хеш-функцій, такими як Pedersen Hash [11], Poseidon використовує до 8 разів менше обмежень на біт повідомлення, що забезпечує значне пришвидшення роботи. Варто згадати, що на початку 2020 року системи з нульовим розголошенням мають досить велику популярність, зумовленою фактами викладеними у першому розділі. Можна виділити 3 системи доказу з нульовим розголошенням: zk-SNARKs (про яку вже йшлось вище), Bulletproofs [12] , zk-STARKs [13]. Перші два протоколи вже використовуються у багатьох системах на практиці. Попри те, що zk-STARKs ще не отримав настільки широкого застосування у реальних системах, він є найбільш перспективним з точки зору продуктивності та постквантової безпеки. Ці три системи використовують дві досить різні схеми опису, що зумовлює різний розмір доказів та час генерації:

– R1CS (англ. Rank-1 quadratic constraints) яка вже була оглянута нами вище. Використовується у системах SNARKs та Bulletproofs.

– Метричний показник AET, який використовується в zk-STARK. Виражається як сукупність внутрішніх станів, пов'язаних між собою поліноміальними рівняннями степеня d. Стан складається з w двійкових елементів поля $GF(2^n)$ і зазнає T перетворень. Генерація доказе приблизно пропорційна добутку $w \cdot d \cdot T$, де n має бути 32 або більше.

Однією з цілей розробників Poseidon було створення сімейства хеш-функцій, які б однаково добре підходили і для R1CS, і для AET. Функція Poseidon базується на стратегії HADES і застосуванням функції губки. Механізми, що вже були наведені вище.

Застосування функції губки Як вже було доведено Бертоні у роботі [9] внутрішня перестановка P у N-бітній функціїї губки ($N = r + c$. де r - бітова швидкістю, та — потужність) змодельована як випадково обрана перестановка, не буде відрізнятись від випадкового оракула до $2^{c/2}$ звернень до P. Функція губки з потужністю , має рівень захисту $2^{c/2}$ до колізійної атаки та атаки знаходження прообразу другого типу. Маючи перестановку розміру N і бажаний рівень надійності s, ми можемо хешувати $r = N - 2s$ біт за один виклик P. З огляду на це, обирається кількість раундів внутрішньої перестановки, щоб бути впевненими що наші перестановки не показують залежності між собою, протягом 2^M запитів, де M це бажаний рівень надійності. Іншими словами, така перестановка не буде відрізнятись від абсолютно випадкової, для 2^M елементів.

Через те, що повідомлення, відповідно до специфікації функції губки, спочатку заповнюється нулями, кількість блоків-повідомлень буде кратна r. У випадку функції "Poseidon-256"(аналог Starkad-256), ми використовуємо $N \leq 4 \cdot M$ (де M рівень захисту) та $N = n \cdot t \leq 1024$ Цей вибір дозволяє нам оброблювати більшу кількість вхідних бітів, ніж, наприклад, у SHA-256 (512 біт) [14].

Перестановки у Poseidon Перестановки у Poseidon відбуваються

шляхом застосування стратегії HADES. Шар замішування визначається множенням фіксованої $t \times t$ MDS-матриці (від англ. Maximum Distance Separable). Кількість раундів $R = 2 \cdot R_f + R_p$ залежить від вибору S-блоку та параметрів t і n. У нашому випадку, застосовуються такі S-блоки:

– Кубічний S-блок. $S\text{-}box(s) = x^3$. - нагадаємо, що кубічний S-блок, це бієкція у $GF(2^n)$ якщо n - непарне, та бієкція у $GF(p)$, якщо $p = 2 \bmod 3$. Далі, такі перестановки, будемо називати "$x^3 - Poseidon$".

– S-блок. $S\text{-}box(x) = x^5$ - нагадаємо, що x^5 це бієкція в $GF(2^n)$ якщо $(2^n) \neq 1 \bmod 5$, і це бієкція у $GF(p)$ якщо $p \neq 1 mod 5$. Далі, такі перестановки, будемо назвати "$x^5 - Poseidon$".

– обернений S-блок. $S\text{-}box(x) = x^{-1}$. У цьому випадку, ці перестановки будемо називають "$x^{-1} - Poseidon$".

Висновки до розділу 2

У даному розділі, ми познайомилися з новою, та доволі перспективною функцією геш-функцією Poseidon, як має хорошу перспективу для застосування у доказах з нульовим розголошенням. Ми розглянули також допоміжні конструкції, за допомогою яких вона була побудована, а саме, блоковий алгоритм Hades, та конструкцію губки. Ці знання, допоможуть нам у наступному розділі оцінити рівень надійності запропонованої геш-функції, та зробити висновки щодо доцільності її використання у майбутньому.

3 ОЦІНКА НАДІЙНОСТІ ГЕШ-ФУНКЦІЇ ПРИ РІЗНИХ ТИПАХ АТАК

У даному підрозділі ми розглянемо більше конкретно надійність Poseidon з різними раундовими функціями, та оцінимо її стійкість до диференціального (різницевої) та лінійного криптоаналізу при різному виборі s-блоків у базовому алгоритмі HADES.

Автори алгоритму Hades аналізуючи стійкість цього алгоритму до різних типів атак, зокрема алгебраїчних та статистичних помились. І у роботі в якій було представлено геш-функцію POSEIDON, ці помилки було повторено. Оскільки алгоритми Hades та POSEIDON набувають все більшої популярності у технології блокчейн, питання побудови обґрунтованих оцінок їх криптографічної стійкості є надзвичайно актуальним. Використання цих алгоритмів з параметрами, які не відповідають заявленому рівню стійкості, може призвести як до втрати криптовалюти її власниками, так і до порушення заявленого рівня анонімності при виконанні транзакцій або при роботі зі смарт-контрактами.

Зокрема, ми ставимо перед собою наступні завдання у цьому підрозділі:

1) описати помилки, допущені у оригіній роботі [8];

2) оцінити рівень безпеки функції перестановки проти різних атак;

3) оцінити кількість раундів, достатніх для досягнення заданого рівня безпеки;

4) оцінити кількість обмежень на біт, для безпечної перестановки.

3.1 Формалізація математичних конструкцій.

Формалізуємо деякі поняття, які вже були наведені у попередніх розділах, таким чином, щоб нам було зручніше з ними працювати.

Спочатку, наведемо математичну модель перестановки Hades, на якій базується Poseidon.

Нехай p велике просте число, l його бітова довжина, $l \approx log\, p$. В нашому випадку, $l \approx 753$.

Ми визначаємо бієкцію $s : F_p \to \mathrm{F}_p$ як $s(x) = x^u \bmod p$, де $(u, p-1) = 1$. Тепер оберемо два S-блоки для аналізу, першим у нас буде один із блоків, запропонований авторами HADES, а саме:

$$s(x) = \begin{cases} x^{-1} mod\, p & \text{, якщо } x \neq 0 \\ 0 & \text{, інакше.} \end{cases} \tag{3.1}$$

Для зручності, ми будемо позначати його як $S\text{-}box(x) = x^{-1}$

Інший S-блок, який ми будемо аналізувати буде мати наступний вигляд: $s(x) = x^{13} mod\, p$.. Для зручності, ми будемо позначати його аналогічно до попереднього, $\mathrm{S\text{-}box}(x) = x^{13}$

Наведені далі у цій секції означення справедливі для обох випадків, поки не буде сказано протилежного.

Для деякого $t \in N$ визначимо значення x, $C \in (F_p)^t$, як $x = (x_1, ..., x_t)$, $C = (c_1, ...c_t)$, де $x_i, c_i \in \mathrm{F}_p$. для $i = \overline{(1, t)}$. Тепер, для $x \in (\mathrm{F}_p)^t$, визначено два відображення: $S^{full} : (F_p)^t \to (\mathrm{F}_p)^t$ та $S^{part} : (F_p)^t \to (\mathrm{F}_p)^t$ як: $S^{full}(x) = (s(x_t), .., s(x_1))$ і $S^{part}(x) = (x_t, ..., s(x_1))$.

MDS-матриця $A : (F_p)^t \to (\mathrm{F}_p)^t$, розмірності $t \times \mathrm{t}$.

Визначимо раундові функції для перестановки Hades. Як ми вже знаємо, вони можуть бути двох видів: з повним шаром S-блоків, які ми визначаємо як $f_c^{full} : (F_p)^t \to (\mathrm{F}_p)^t$, де для довільного C $\in (\mathrm{F}_p)^t$: $f_c^{full}(x) = A \circ \mathrm{S}^{full}(x * C)$. І аналогічно, ми визначаємо функцію для

випадку з не повним шаром S-блоків. $f_c^{part} : (F_p)^t \to (\mathrm{F}_p)^t$, де для довільного $\mathrm{C} \in (\mathrm{F}_p)^t$: $f_c^{full}(x) = A \circ \mathrm{S}^{part}(x * C)$. де $(x * C) = (x_t + c_t, ..., x_1 + t_1)$, а знак $+$, є додаванням за модулем p

Означення 3.1. Ми будемо стверджувати що перестановка відноситься до типу Hades, з параметрами p, t, u, r_full, r_part, та сімейством функції $H_c^{p,t,u,r_full,r_part} : (F_p)^t \to (\mathrm{F}_p)^t$, параметризованої по $C = (C_1, ...C_{2r_{full}+r_{part}})$, $C_i \in (F_p)^t$, яка визначається як:

$$H_C^{(p,t,u,r_{full},r_{part})}(x) = f_{C_{2r_{full}+r_{part}}}^{full} \circ ... \circ f_{C_{r_{full}+r_{part}+1}}^{full} \circ f_{C_{2r_{full}+r_{part}}}^{part} \circ ... \circ f_{C_{r_{full}+1}}^{part} \circ f_{C_{r_{full}}}^{full} \circ f_{C_1}^{full}(x)$$

Якщо параметри p, t, u, r_full, r_part задані, ми будемо позначати це просто як H_c для спрощення математичних викладок.

3.2 Оцінка надійності у моделі випадкового оракулу

Рівень надійності функції губки здебільшого залежить від рівня надійності її внутрішньої перестановки, в нашому випадку Hades. Ми вже давали посилання на роботу, у якій було доведено, що якщо внутрішня перестановка моделюється як випадково обрана перестановка, то функція губки невідмінна від випадкового оракула до $2^{c/2}$ її викликів.

Ми встановимо бажаний рівень захисту $\lambda = 128$ біт. Отже, потужність в функціїї губки має задовільняти нерівності $\frac{c}{2} \geq 128$. Для зручності ми беремо $\approx 2l(p)$. Або, більш точніше $c = 1504$. (184 байти) $r = 752$ (92 байти). Таке значення потужності є надлишковим для рівня надійності $\lambda = 128$, проте воно досить зручне при використані на практиці. Рівень надійності $\lambda = 128$ означає що ми маємо довести що рівень безпеки внутрішньої перестановки Hades не менше 128 біт. Ми будемо використовувати цю вимогу, щоб знайти кількість раундів цієї перестановки.

3.3 Застосування диференціального криптоаналізу.

Основний результат з теорій диференціального криптоаналізу, який застосовують автори Hades [8] і який дозволяє виразити стійкість алгоритму через параметр, що характеризує *S-блок*, індекс галуження та кількість раундів, є справедливим лише за умови, що раунди з повним набором S-блоків ідуть підряд, не перемежаючись з раундами, які містять зменшений набір S-блоків. Тому для алгоритму Hades застосування цього результату є некоректним.

У подальшому, ми будемо використовувати ідеї диференціального криптоаналізу, запропоновані у роботі [15]

Щоб побудувати оцінку надійності проти диференціального криптоаналізу, ми розглянемо Hades як r-раундовий блоковий шифр . Далі ми будемо використовувати наступні результати

Означення 3.2. Блок шифр E з r-раундовими функціями: $f : M \times K \to M$ (де M є абелевою групою, тобто має операцію *, і 0 нейтральний елемент) називається *марківським шифром*, якщо: $\forall x, \alpha, \beta$:

$$\frac{1}{|K|}\sum_{k\in K}\delta\Big(f\left(k,x*\alpha\right)*f\left(k,x\right)^{-1},\beta\Big)=\frac{1}{|K|}\sum_{k\in K}\delta\Big(f\left(k,\alpha\right)*f\left(k,0\right)^{-1},\beta\Big)$$

де δ - символ Кронекера.

$$\delta(x,y)=\begin{cases}1 & \text{, якщо } x=y\\ 0 & \text{, інакше.}\end{cases}$$

Зауваження. Це визначення можна узагальнити для випадку, коли раундові функції, є різними для різних раундів.

Означення 3.3. Індексом галуження лінійного оператора матриці

$A : (F_p)^t \to (F_p)^t$ розмірності $t \times t$, називається величина

$$br(A) = \min_{x \in (F_p)^t} (wt(ax) + wt(x)) \tag{3.2}$$

де wt є вагою Хемінга вектора, відповідного вектора елементів t

Якщо A це MDS-матриця, тоді індекс галуження лінійного оператора є максимально можливим, і рівним $br(A) = t + 1$. Наприклад, у нашому випадку $t = 3$, і $br(A) = 4$

Лема 3.1. *Блочний шифр наведений вище є марківським шифром.*

Це ствердження може бути доведено безпосередньо, перевіривши раундові функції

Лема 3.2. *Для марковського шифра його захищеність від диференціального криптоаналізу оцінюється зверху, зі значенням* Δ^b *де* $+$ *- додавання в полі, додавання зворотного елемента в полі* b *-*

$$\Delta = \Delta(s) = \max_{\alpha,\beta \in F_p^*} \frac{1}{p} \sum_{x \in F_p} \delta\big(s(x+\alpha) - s(x), \beta\big)$$

кількість активних s-блоків у всіх раундах.

Лема 3.3. *Кількість активних S-блоків у двох послідовних раундах з раундовою функцією не менше індекса галуження лінійного оператора* $br(A)$*.*

У нашому випадку A є MDS-матрицею розміру $t \times t$, отже $br(A) = 4$

Лема 3.4. *Очевидно що кількість активних S-блоків в усіх раундах, не менша від кількості активних S-блоків у раундах з повним шаром S-блоків.*

Лема 3.5. *Унаслідок* ***Леми 3.2*** *та* ***Леми 3.3****. Кількість активних S-блоків* H_c*, не менша ніж* $4 \cdot 2 \cdot \left[\frac{r_{full}}{2}\right] = 8 \cdot \left[\frac{r_{full}}{2}\right]$

Для випадку S-box$(x) = x^{-1}$

Теорема 3.1. *Нехай,*

$$s(x) = \begin{cases} x^{-1} mod\, p & \text{, якщо } x \neq 0 \\ 0 & \text{, інакше.} \end{cases}$$

тоді, ми стверджуємо що $\Delta \leq \frac{4}{p}$

Доведення. Спочатку, ми доведемо що для будь-якого $\alpha, \beta \in F_p^*$ рівняння: $s(x+\alpha) - s(x) = \beta$ має не більше ніж 4 рішення у полі F_p. Розглянемо 3 наступні варіанти:

1) x = 0. Тоді, ми можемо записати це рівняння як $s(\alpha) - s(0) = \beta$, або $\alpha^-1 = \beta$. Отже, ми маємо як мінімум один корінь x = 0.

2) $x = -\alpha$. Тоді, ми можемо переписати це рівняння як $s(0) - s(-\alpha) = \beta$, або $\alpha^-1 = \beta$. Отже, ми маємо ще один корінь x = 0.

3) $x(x+\alpha) \neq 0$ Тоді ми можемо переписати це рівняння як $(x+\alpha)^{-1} - (x)^{-1} = \beta$ або $x - (a+\alpha) = \beta x(x+\alpha)$. Після множення правої і лівої сторони на $x(x+\alpha)$ ми отримуємо: $\beta x^2 + \alpha\beta x = 0$ Ліва частина цього полінома має степінь 2 над полем F_p. Такий поліном не може мати більше ніж 2 корені у F_p, Отже, у цьому випадку, не може бути більш ніж 2 рішення.

Випадки 1 та 2 дали нам по 1 рішенню. Отже, для будь-яких $\alpha, \beta \in F_p^*$ рівняння: $s(x+\alpha) - s(x) = \beta$ не може мати більш як 4 рішення. □

Для випадку *S-box* $(x) = x^{13}$

Теорема 3.2. *Нехай,* $s(x) = x^{13} \, mod\, p$. *тоді, ми стверджуємо що* $\Delta \leq \frac{12}{p}$

Доведення. Спочатку, ми доведемо що для будь якого $\alpha, \beta \in F_p^*$ рівняння: $s(x+\alpha) - s(x) = \beta$ має не більше ніж 12 рішення у полі F_p. Справді, запишемо це рівняння як $(x+\alpha)^{13} - x^{13} = \beta$. Після розкриття дужок, ми будемо мати: Ліва частина, це поліном степені 12 над F_p. Такий поліном не може мати більше ніж 12 коренів в F_p, отже і рівняння не може мати більше ніж 12 рішень. З цього твердження, ми отримуємо що сума

$$\sum_{i=1}^{13}\binom{13}{i}x^{13-i}\alpha^{i}-\beta=0\,.$$

правої частини у **Лемі 3.1** не може бути більшою ніж 12 для будь якого $\alpha, \beta \in F_p^*$, отже теорема доведена. □

Для випадку S-box$(x) = x^{-1}$

Теорема 3.3. *Нехай,* r_{full} *є парним числом.*

$$s(x) = \begin{cases} x^{-1} mod\, p & \text{, якщо } x \neq 0 \\ 0 & \text{, інакше.} \end{cases}$$

, $A : (F_p)^3 \to (F_p)^3$ *це MDS матриця розмірності* $t \times t$*. Тоді, верхня оцінка рівня надійності блокового шифру* H_c *проти диференціального криптоаналізу оцінюються значенням:* $\Delta^b = \left(\frac{4}{p}\right)^{4_{r\,full}}$

Доведення. Є очевидним, використовуючи вищенаведені леми та теорему, та той факт що $\left[\frac{r_{full}}{2}\right] + \left[\frac{r_{full}}{2}\right] = r_{full}$ □

Ми обрали рівень надійності = 128. Тоді, $\left(\frac{4}{p}\right)^{4_{r\,full}} < 2^{-128}$ Але, для покращення надійності та наближення до теоретичного значення, ми покладаємо: $\left(\frac{4}{p}\right)^{4_{r\,full}} < 2^{2\cdot -128} = 2^{-256}$ Звідки можемо бачити, що навіть у випадку лише двох повних раундів (мінімально можливої кількості r_{full}), рівень захисту проти диференціального криптоаналізу, буде $\left(\frac{4}{p}\right)^{4} \approx 2^{-3000} \ll 2^{256}$

Це означає, що перестановка є безпечною навіть у випадку 2 повних раундів. Додавши два додаткових раунди, та отримавши повний шар S-перестановок (як це і було запропоновано в алгоритмі Poseidon), та отримавши 4 повні раунди (два раунди на початку, два раунди в кінці, щоб уникнути атаки "зворотнього шляху"). Таким чином, достатня кількість раундів з повним шаром S-перестановок становить 4.

Для випадку S-box$(x) = x^{13}$

Теорема 3.4. *Нехай,* r_{full} *є парним числом.* $s(x) = x^{13} mod\, p$, $A : (F_p)^3 \to (F_p)^3$ *це MDS матриця розмірності* $t \times t$. *Тоді, верхня оцінка рівня надійності блокового шифру* H_c *проти диференціального криптоаналізу оцінюються значенням:* $\Delta^b = \left(\frac{12}{p}\right)^{4_{r\,full}}$

Доведення. Аналогічно до випадку з *S-box*$= x^{-1}$ є очевидним, використовуючи вищенаведені леми та теорему, та той факт що

$$\left[\frac{r_{full}}{2}\right] + \left[\frac{r_{full}}{2}\right] = r_{full}$$

□

Ми вважаємо що блоковий шифр є практично стійким до диференціального криптоаналізу, якщо оцінка його стійкості, не перевищує 2^{-N}, де N - розмір блоку. Ми вже показували, що у нашому випадку для конструкції губки, максимальний рівень захисту $l(p) \approx log\, p$. Отже, найслабшу вимогу, ми можемо сформулювати як: $\left[\frac{r_{full}}{2}\right] + \left[\frac{r_{full}}{2}\right] < 2^{-log\, p}$, але задля покращення рівня надійності, та наближення його до теоретичної оцінки надійності, ми будемо стверджувати: $\left[\frac{r_{full}}{2}\right] + \left[\frac{r_{full}}{2}\right] < 2^{-2N} \approx 2^{-6log\, p}$ звідки маємо нерівність $r_{full} \geq 2$. Додавши 2 повниї екстра раунди, ми отримуємо $r_{full} = 3$, але так як за умовою алгоритму HADES, кількість r_{full} має бути парною, приймаємо $r_{full} = 4$. В результаті, ми можемо стверджувати що кількість раундів з повним набором S-блоків, достатню для гарантування стійкості алгоритму до диференціального криптоаналізу, буде рівна 8.

3.4 Застосування лінійного криптоаналізу

Одна з помилок є аналогічною до тої, що описана вище і стосується різницевого криптоаналізу. Друга помилка полягає у тому, що результати, які застосовуються у для побудови оцінок стійкості, є справедливими лише для випадку "класичного"лінійного криптоаналізу, який не можна застосовувати для алгоритмів, побудованих з використанням операцій у простому скінченному полі. До алгоритмів типу Hades можна

застосовувати лише узагальнений лінійний криптоаналіз, описаний у роботі [16]. У подальшому, ми будемо використовувати ідеї диференціаонального криптоаналізу, запропоновані у роботі [16]. Згідно цієї роботи, показником практичної стійкості шифру (відносно операції додавання у скінченному полі) до узагальненого лінійного криптоаналізу є величина:

$\max_{\chi,\rho,\in F_p} ELP^E = L^b$, де величина b є кількістю активних S-блоків, а параметр L залежить від S-блоку і визначається так:

$$\mathrm{L} = \mathrm{L(x)} = \max_{\chi,\rho,\in F_p} \left| \frac{1}{p} \sum (\overline{\chi}(x), \rho(s(x))) \right|^2 \quad (3.3)$$

Фунції χ та ρ є адитивними характерами поля F_p, (тобто, характерами адитивної групи цього поля)

Величина b визначається так само, як і у попередньому розділі. $b = t \cdot r_{full}$, за умовою що r_{full} є парним числом.

Для випадку S-box$(x) = x^{-1}$

Теорема 3.5. *Нехай,* r_{full} *є парним числом.*

$$s(x) = \begin{cases} x^{-1} mod\, p & \text{, якщо } x \neq 0 \\ 0 & \text{, інакше.} \end{cases} \quad (3.4)$$

, Тоді $L(s) \leq \dfrac{16}{p}$

Доведення. Для початку, оцінимо наступну величину:

$$\sum_{x \in F_p} (\overline{\chi}(x), \rho(s(x))) = \sum_{x \in F_p} (\overline{\chi}(x), \rho(x^{-1})) + 1$$

Зазначимо, що група $(F_p, +)$ є циклічною (з генератором $g = 1$),

тому відповідна група характеристик $(F_p, \times)$ теж є циклічною. Нехай Ψ є генератором групи $(F_p, \times)$. Тоді будь-який елемент цієї групи, зокрема характери $\overline{\chi}$ та ρ , можуть бути представлені як: $\overline{\chi} = \Psi^{\alpha}$, $\rho = \Psi^{\beta}$ для деяких $0 \leq \alpha$ та $\beta \leq p-1$ Тоді,

$$\overline{\chi}(x)\rho(x^{-1}) = \Psi(x)^{\alpha}\Psi(x^{-1}) = \Psi(\alpha\, x) \cdot \Psi(\beta\, x^{-1}) = \Psi(\alpha\, x + \beta\, x^{-1})$$

Використовуючи той факт, що характер Ψ - гомоморфізм.

$$\sum_{x \in F_p} (\overline{\chi}(x), \rho(s(x))) = \sum_{x \in F_p} \Psi(\alpha\, x + \beta\, x^{-1})$$

Використовуючи теорему про суму Клоостермана [17], Теорема 1.5, ми отримуємо:

$$\sum_{x \in F_p} \Psi(\alpha\, x + \beta\, x^{-1}) = 2 \cdot \sum_{x_1, x_2 \in F_p} \Psi(x_1 + x_2\, x^{-1}) \leq 2 \cdot 2 \cdot \sqrt{p} = 4\,\sqrt{p}$$

Використовуючи рівняння для оцінки L, отримуємо:

$$\begin{aligned} L = L(x) = \max_{\chi, \rho, \in F_p} \left| \frac{1}{p} \sum \overline{\chi}(x), \rho(s(x)) \right|^2 = \\ = \max_{\alpha, \beta, \in F_p} \left| \frac{1}{p} \left(\sum_{x \in F_p} \Psi(\alpha\, x + \beta\, x^{-1}) + 1 \right) \right|^2 = \left| \frac{1}{p} \cdot 4\sqrt{p} \right|^2 = \frac{16}{p} \end{aligned} \tag{3.5}$$

Тепер кількість раундів з повним набором S-блоків, при якій алгоритм є гарантовано стійким до лінійних атак, можна легко обчислити з нерівності:

$$\frac{16}{p}^{4r_{full}} \leq 2^{-256} \tag{3.6}$$

звідки отримуємо, $r_full = 2$. Отже, навіть у випадку коли ми маємо лише два повних раунди, надійність проти лінійного криптоаналізу буде становити:

$$\frac{16}{p}^{4r_{full}} \approx 2^{-2996} << 2^{-256}$$

□

Отже, додавши два раунди з повними S-блоками (як було запропоновано в роботі Poseidon), отримуємо 4 повних раунди. 2 раунди на початку, 2 в кінці, щоб уникнути атаки зворотнього шляху. Тому, ми можемо стверджувати що кількість раундів з повними S-блоками, необхідна для захисту від лінійних атак - 4.

Для випадку S-box$(x) = x^{13}$

Теорема 3.6. *Нехай,* r_{full} *є парним числом.*

$$s(x) = x^{13} mod\, p$$

Тоді $L(s) \leq \dfrac{144}{p}$

Доведення. Для початку, оцінимо наступну величину:

$$\sum_{x \in F_p} (\overline{\chi}(x), \rho(s(x))) = \sum_{x \in F_p} (\overline{\chi}(x), \rho(x^{13}))$$

Зазначимо, що група $(F_p, +)$ є циклічною (з генератором $g = 1$), тому відповідна група характеристик $(F_p, \times)$ теж є циклічною. Нехай Ψ є генератором групи $(F_p, \times)$. Тоді будь-який елемент цієї групи, зокрема характери $\overline{\chi}$ та ρ , можуть бути представлені як: $\overline{\chi} = \Psi^{\alpha}$, $\rho = \Psi^{\beta}$ для деяких $0 \leq \alpha$ та $\beta \leq p - 1$ Тоді,

$$\overline{\chi}(x)\rho(x^{13}) = \Psi(x)^{\alpha}\Psi(x^{13}) = \Psi(\alpha\, x) \cdot \Psi(\beta\, x^{13}) = \Psi(\alpha\, x + \beta\, x^{13})$$

Використовуючи той факт, що характер Ψ - гомоморфізм. Можемо записати у наступному вигляді

$$\sum_{x \in F_p} (\overline{\chi}(x), \rho(x^{13})) = \sum_{x \in F_p} \Psi(\alpha\, x + \beta\, x^{13})$$

Використовуючи Теорему Вейля [18], Теорема 5.38, ми отримуємо:

$$\sum_{x \in F_p} \Psi(\alpha\, x + \beta\, x^{-1}) \leq (deg(\alpha\, x + \beta x^{13}) - 1) \cdot \sqrt{p} = 12\sqrt{p}$$

Використовуючи рівняння для оцінки L, отримуємо:

$$L = L(x) = \max_{\chi,\rho,\in F_p} \left| \frac{1}{p} \sum (\overline{\chi}(x), \rho(s(x))) \right|^2 = \\ = \max_{\alpha,\beta,\in F_p} \left| \frac{1}{p} \sum_{x \in F_p} \Psi(\alpha\, x + \beta\, x^{13}) \right|^2 = \left| \frac{1}{p} \cdot 12\sqrt{p} \right|^2 = \frac{144}{p} \tag{3.7}$$

□

Тепер кількість раундів з повним набором S-блоків, при якій алгоритм є гарантовано стійким до лінійних атак, можна легко обчислити з нерівності:

$$\frac{144}{p}^{4r_{full}} \leq 2^{-2\,N} = 2^{-6\,p}$$

звідки отримуємо, $r_{full} = \dfrac{6 \cdot 753}{4 \cdot 745} = 2$ Отже, додавши два раунди з повними S-блоками (як було запропоновано в роботі Poseidon), отримуємо 4 повних раунди. 2 раунди на початку, 2 в кінці, щоб уникнути атаки зворотнього шляху. Тому, ми можемо стверджувати що кількість раундів з повними S-блоками, необхідна для захисту від лінійних атак - 4.

Висновки до розділу 3

У цьому розділі, ми провели лінійний та диференціальний (різницевий) криптоаналіз для блокового шифру, що застосовується для побудови геш-функції Poseidon. Зокрема, ми перевірили це для двох видів S-блоків. *S-block* $= x^{-1}$, запропонованого авторами оригінального алгоритму HADES, та та *S-block* $= x^{13}$. З наших результатів, ми вперше отримали конретні та математично обрунтовані оцінки, для безпечного використання цих S-блоків. Дані результати, можуть бути використані для побудови нових, чи покращення вже існуючиї Геш-Функцій.

ВИСНОВКИ

У цій роботі ми оглянули проблематику анонімності у криптовалютах та розглянули існуючи способи покращення анонімності при проведенні транзакції у криптовалютах, зокрема, використання zk-SNARK протоколів. Також дослідили запропоновану геш-функцію, яка була розроблена для використання у доведеннях без розголошення.

Ми окреслили проблему, а саме ненадійність математичної оцінки блочного шифрування HADES, яке було представлене авторами. Також ми спробували дати свою, математично та науково обґрунтовану оцінку щодо надійності запропонованої геш-функції Poseidon, побудовану на ідеї блокового шифру HADES.

Ми отримали чіткі вирази для різних S-блоків, для:

1) Оцінки рівня безпеки функції перестановки проти різних атак;

2) Оцінки кількості раундів, достатніх для досягнення заданого рівня безпеки;

3) Оцінки кількості обмежень на біт для безпечної перестановки.

Наведені у нашій роботі результати можна також використовувати для визначення таких параметрів алгоритму (як, вибір кількості раундів та блоків заміни), які гарантують необхідний рівень стійкості алгоритму до статистичних атак. Та, враховуючи їх, будувати шляхом зміни цих параметрів нові, можливо, більш надійні алгоритми, які будуть мати науково обґрунтовану стійкість.

ЛІТЕРАТУРА

[1] Satoshi Nakamoto. *Bitcoin: Apeer-to-peer electronic cash system*. 2008.

[2] European Parliament та of the Council. *Directive (EU) 2018/843*. 2018.

[3] Surae and Monero Core Team. *Improving Obfuscation in the CryptoNote Protocol*. 2015.

[4] Ben-Sasson and Alessandro Chiesa and Christina Garman and Matthew Greea. *Zerocash: Decentralized Anonymous Payments from Bitcoin*. 2014.

[5] Eli Ben-Sasson Technion and Alessandro Chiesa. *Succinct Non-Interactive Zero Knowledge*. 2018.

[6] Eli Ben-Sasson and Iddo Bentov and Yinon Horesh and Michael Riabzev. *Scalable, transparent, and post-quantum secure computational integrity*. 2018.

[7] G. Bünz B and Bootle J and Boneh D and Poelstra A and Wuill P. Maxwell. *Bulletproofs: Short proofs for condential transactions and more*. 2018.

[8] Lorenzo Grassi and Reinhard Lüftenegger and Christian Rechberger. *On a Generalization of Substitution-Permutation Networks: The HADES Design Strategy*. 2018.

[9] Guido Bertoni and Joan Daemen and Micheale Peeters and Gilles Van Assche. *Cryptographic sponge functions*. 2008.

[10] Lorenzo Grassi and Daniel Kales and Dmitry Khovratovich and Arnab Roy and Christian Rechberger. *Starkad and Poseidon: New Hash Functions for Zero Knowledge Proof Systems*. 2019.

[11] Hopwood D and Bowe S and Hornby T. *Zcash protocol specication Version 2019*. 2019.

[12] P Bünz B. and Bootle J. and Boneh D. and Poelstra A. and Wuille. *Bulletproofs Short proofs for condential transactions and more*. 2018.

[13] Ashur T. and Dhooghe S. *Marvellous a stark-friendly family of cryptographic primitives*. 2018.

[14] ANDREW W. APPEL. *Verification of a Cryptographic Primitive: SHA-256*. 2009.

[15] Lorenzo Grassi. *Mixture Dierential Cryptanalysis: a New Approach to Distinguishers and Attacks on round-reduced AES*. 2017.

[16] Thomas Baigneres, Jacques Stern та Serge Vaudenay. *Linear Cryptanalysis of Non Binary Ciphers*. 2017.

[17] Rudolf Lidl. *Introduction to Finite Fields and their Applications*. 2012.

[18] Quoc P. Ho. *Some Notes on Trigonometric Sums*. 2014.